Package ‘minimapR’

March 30, 2025

Title Wrapper for 'minimap2'

Version 0.0.1.1

Date 2025-03-28

Maintainer Jake Reed <`hreed.ga@gmail.com`>

Description

Wrapper for 'Minimap2'. 'Minimap2' is a very valuable long read aligner for the Pacbio and Oxford Nanopore Technologies sequencing platforms. 'minimapR' is an R wrapper for 'minimap2' which was developed by Heng Li <me@liheng.org>.
*SPECIAL NOTES
1. Examples can only be run from 'GitHub' installation.
2. 'conda' or 'mamba' must be used to install 'minimapR' on your system.
Li, Heng (2018) <doi:10.1093/bioinformatics/bty191> ``Minimap2: pairwise alignment for nucleotide sequences''.

License MIT + file LICENSE

Depends R (>= 4.4.0), Rsamtools, pafr

URL `https://github.com/jake-bioinfo/minimapR`

Encoding UTF-8

RoxygenNote 7.3.2

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Jake Reed [aut, cre] (<`https://orcid.org/0000-0002-8384-3593`>),
Pradyun Pulipaka [ctb]

Repository CRAN

Date/Publication 2025-03-30 16:10:02 UTC

Contents

`minimap2` *minimap2*

Description

This function is a wrapper for the command line tool minimap2. minimap2 is a long read sequencing alignment tool that is used to align long reads to a reference genome.

Usage

```
minimap2(
  reference,
  query_sequences,
  output_file_prefix = "minimap2_out",
  a = TRUE,
  preset_string = "map-hifi",
  threads = 1,
  return = FALSE,
  verbose = TRUE,
  ...
)
```

Arguments

`reference`	Reference genome to align the query sequences
`query_sequences`	Query sequences to align to the reference genome
`output_file_prefix`	Output file to save the alignment results
`a`	Logical value to use the preset string with the -a flag
`preset_string`	Preset string to use with the -x flag
`threads`	Number of threads to use
`return`	Logical value to return the alignment results
`verbose`	Logical value to print progress of the installation
`...`	Additional arguments to pass to minimap2

Value

This function returns the line needed to add minimap2 to PATH

Examples

```
## Not run:
reference <- system.file("extdata/S288C_ref_genome.fasta.gz", package = "minimapR")
query_sequences <- system.file("extdata/yeast_sample_hifi.fastq.gz", package = "minimapR")
# Warning: not setting output_file_prefix will generate the output in the current working directory
bam_out <- minimap2(reference,
 query_sequences,
 threads = 4,
 preset_string = "map-hifi",
 return = TRUE,
 verbose = TRUE)

## End(Not run)

## Not run:
reference <- system.file("extdata/GRCh38_chr1_130k.fa.gz", package = "minimapR")
query_sequences <- system.file("extdata/ont_hs_sample.fastq.gz", package = "minimapR")
# Warning: not setting output_file_prefix will generate the output in the current working directory
bam_out <- minimap2(reference,
 query_sequences,
 threads = 4,
 preset_string = "map-hifi",
 return = TRUE,
 verbose = TRUE)

## End(Not run)
```

`minimap2_check` *minimap2_check*

Description

Check if minimap2 is installed

Usage

```
minimap2_check(return = TRUE)
```

Arguments

`return`	Logical value to return the path of minimap2

Value

If minimap2 is installed, this function returns the path of minimap2 (character).

Examples

```
minimap2_check(return = TRUE)
```

`minimap2_installation` *minimap2_installation*

Description

This function prints installation instructions specific to the user's operating system.

Usage

```
minimap2_installation(verbose = TRUE)
```

Arguments

`verbose`	Logical value to print progress of the installation

Value

This function returns the path of the installed 'minimap2' tool (character).

Examples

```
## Not run:
minimap2_installation(verbose = FALSE)

## End(Not run)
```

`mm2_install` *minimap2_install*

Description

Install `minimap2` using conda or mamba. Conda or mamba is required for installation.

Usage

```
mm2_install(verbose = TRUE)
```

Arguments

`verbose`	Logical value to print progress of the installation. Default is `TRUE`. If set to `FALSE`, it will suppress the output messages during installation.

Value

If '`minimap2`' is not installed, this function installs it on windows, linux or macOS.

Examples

```
## Not run:
mm2_install(verbose = TRUE)

## End(Not run)
```

`samtools_check` *samtools_check*

Description

Check if samtools is installed

Usage

```
samtools_check(return = TRUE)
```

Arguments

`return`	Logical value to return the path of samtools

Value

If '`samtools`' is installed, this function returns the path of samtools (character).

Examples

```
samtools_check(return = TRUE)
```

`samtools_install` *samtools_install*

Description

Install samtools with conda

Usage

```
samtools_install(verbose = TRUE)
```

Arguments

`verbose`	Logical value to print progress of the installation

Value

If '`samtools`' is not installed, this function installs it returns the path of the installed `'samtools'` tool (character).

Examples

```
## Not run:
samtools_install()

## End(Not run)
```

Index